DRAFT

Measurement Types
for the
Recycler Ring BPM Front-end

Duane C. Voy

Introduction

The Recycler BPM front-end is designed to provide three measurement algorithms:

- repetitive single– arm once, collect on all subsequent triggers,
- one-shot multiple– arm & trigger once, collect on multiple contiguous turns and
- one-shot single– arm & trigger once, collect on single turn.

Position measurements are activated with an event driven data acquisition scheme described in "Event Driven Data Acquisition for the Recycler Ring BPM Front-end". This scheme allows for up to sixteen acquisitions each capable of executing any one of the three measurement algorithms. Acquisition specifications define the conditions under which acquisitions are to take place as well as the algorithms to be used in collecting the raw measurement data. Each specification contains the measurement algorithm selection, the measurement trigger conditions and the DSP filter selection among others. Any one of the classic flash, closed orbit or turn-by-turn beam position measurements can be made with the correct combination of acquisition specification settings. Unfortunately not all of the setting combinations are legal and it can be difficult to correctly specify a desired measurement. This note will attempt to demystify the process of creating acquisition specifications and collecting beam position data.

Repetitive Single Gate Algorithm

The repetitive single gate measurement is designed to begin when the specified arm event occurs and then collect data synchronously to the beam turn marker each and every time the specified trigger event occurs. So it is repetitive in the sense that it automatically rearms and repeats the position measurement each time the trigger happens. Because it is a single gate measurement it can not collect turn-by-turn data. As indicated in the table below it can collect flash and averaged data.

Filter	Measurement Data
2.5MHzEnsemble	Flash
2.5MhzBunchbyBubch	Flash

2.5MhzNarrowBand	Averaged
UnbunchedEnsemble	N/A
UnbunchedHeadTail	Flash
89KhzNarrowBand	Averaged

This measurement has been called a 'Background' measurement because it will continue collecting data on each trigger, perhaps temporarily interrupted from time to time by other measurements, until it is disabled. The repetitive single gate measurement always places its data into a circular buffer for fast time plotting and sudden beam loss data collection.

The repetitive single gate measurement can be armed by any appropriate Tclk event or automatically. If armed by a Tclk event the measurement will be aborted and then restarted each time the specified Tclk event occurs. Under normal conditions this makes little sense but each time the measurement is aborted a circular buffer entry with 'aborted' status is created. By arming on Tclk it is possible to place a crude form of Tclk tagging into the circular buffer. The automatic arm feature is most appropriate for use with repetitive single gate measurements. In this case the measurement will begin as soon as the configuring acquisition specification is received by the front-end and continue until it is actively disabled by an updated acquisition specification. When the repetitive measurement is disabled a circular buffer entry with 'disabled' status is created to note the end of data collection.

The repetitive single gate measurement can be triggered by Bsync events or periodically. If Bsync triggering is desired the Bsync event rate must be low enough to give the front-end time to collect, process and store each measurement. It is illegal to use Bsync triggering with the 0xC0 or 0xAA revolution markers. (This would represent an attempt to make a new measurement every eleven microseconds!) The front-end has no way of knowing the rates of other Bsync events so it does not try to guard against other possible high rate events. It is possible to trigger this measurement so often that the front-end is completely consumed by the data collection process and becomes unresponsive to ACNet! Can you say Reboot! The periodic trigger feature is most commonly used with repetitive single gate measurements. It is possible to select a background measurement at any rate from 2 to 100 Hz.

One-shot Multiple Gate Algorithm

The one-shot multiple gate measurement is designed to begin when the specified arm event occurs, collect data synchronously to the beam turn marker for the first 2048 turns immediately following the specified trigger event and then go away. So it is a one-shot measurement that collects more than one data sample. Because it is a multiple gate measurement it is capable of collecting turn-by-turn data from which flashes and closed orbits can be derived. The table below indicates the type of data collected with each of the available filtes.

Filter	Measurement Data
2.5MHzEnsemble	Flash, Closed Orbit and Turn-by-turn
2.5MhzBunchbyBubch	Flash, Closed Orbit and Turn-by-turn
2.5MhzNarrowBand	N/A
UnbunchedEnsemble	N/A
UnbunchedHeadTail	Flash, Closed Orbit and Turn-by-turn
89KhzNarrowBand	N/A

This measurement will interrupt the repetitive (background) measurement for the amount of time required to transition from the arm event to the trigger event and throughout the 2048 turn data collection and read out process. This represents dead time for the repetitive measurement. When the repetitive measurement is interrupted in this way a circular buffer entry with ‘aborted’ status is created to note the lapse in data collection.

The one-shot multiple gate measurement can be armed automatically or by any appropriate Tclk event. If automatically armed the measurement will happen only once as soon as the specified trigger event occurs. If armed by Tclk the measurement will happen each time the specified Tclk arm event and specified trigger occurs.

The one-shot multiple gate measurement should be triggered by Bsync events only. Periodic triggers will work but since this is a one-shot measurement, that makes little sense. There must be a minimum of 400 milliseconds between the arrival of the arm event and the specified trigger event to give the front-end time to configure the measurement before the trigger arrives. If the 400 millisecond requirement is not met the trigger event will most likely be missed and the measurement will time out.

One-shot Single Gate Algorithm

The one-shot single gate measurement is designed to begin when the specified arm event occurs, collect data synchronously to the beam turn marker for the first turn immediately following the specified trigger event and then go away. So it is a one-shot measurement that collects a single data sample. Because it is a single gate measurement it can not collect turn-by-turn data. The table below indicates the type of data collected with each of the available filters.

Filter	Measurement Data
2.5MHzEnsemble	Flash
2.5MhzBunchbyBubch	Flash
2.5MhzNarrowBand	Averaged
UnbunchedEnsemble	N/A
UnbunchedHeadTail	Flash
89KhzNarrowBand	Averaged

This measurement will interrupt the repetitive (background) measurement for the amount of time required to transition from the arm event to the trigger event and throughout the single turn data collection and read out process. This represents dead time for the repetitive measurement. When the repetitive measurement is interrupted in this way a circular buffer entry with 'aborted' status is created to note the lapse in data collection.

The one-shot single gate measurement can be armed automatically or by any appropriate Tclk event. If automatically armed the measurement will happen only once as soon as the specified trigger event occurs. If armed by Tclk the measurement will happen each time the specified Tclk arm event and specified trigger occurs.

The one-shot single gate measurement should be triggered by Bsync events only. Periodic triggers will work but since this is a one-shot measurement, that makes little sense. There must be a minimum of 400 milliseconds between the arrival of the arm event and the specified trigger event to give the front-end time to configure the measurement before the trigger arrives. If the 400 millisecond requirement is not met the trigger event will most likely be missed and the measurement will time out.

End.